//what is socekt in Linux?

A socket is just a logical endpoint for communication. They exist on the transport layer. You can send and receive things on a socket, you can bind and listen to a socket. A socket is specific to a protocol, machine, and port, and is addressed as such in the header of a packet.

//what is TCP socket?

TCP socket is a socket using TCP protocol.

//How does it work?

TCP server socket listens for incoming connections.

1. The server creates a socket.

2. Binds the socket to an IP address and port number .

3. Listens for incoming connections.

4. When a client connects to the server, a new socket for is created for communication with the client.

// please create a TCP server listening at myhost="123.121.10.10" and port=2020

See “server.cpp”

// please also create a TCP client sending "Nice to meet you" to the server above

See “client.cpp”

//what is UDP socket?

UDP socket is a socket using UDP protocol.

//How does it work?

Similar to TCP socket, but it doesn't need to confirm the connection.

//please create a Multicast server at myhost="221.121.10.10" and port=24000

// please also create a receiver listening to the multicast server above

// show the Multicast server casts a message "Welcome to my group"

// Show how the receiver read the message

See “multicastserver.cpp” and “multicastreceiver.cpp”

//what is the main difference between TCP and UDP sockets?

UDP is faster and do not need to make sure the receiver received the data.

//what is kernel bypass?

Instead of using the functions of the kernel, use some program to get the same functions.

//show one example of kernel bypass in socket operation.